UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/692,115	10/23/2003	Cary Lee Bates	CA920020065US1	6581

46073   7590   09/25/2007

IBM CORPORATION (VE)
C/O VOLEL EMILE
P. O. BOX 162485
AUSTIN, TX 78716

EXAMINER
WEI, ZHENG

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
09/25/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

	Application No.	Applicant(s)
Office Action Summary	10/692,115	BATES ET AL.
	Examiner	Art Unit	
	Zheng Wei	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _06 July 2007_.

2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-23_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-23_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _06 July 2007_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

DETAILED ACTION

Remarks

1.   This office action is in response to the amendment filed on 07/06/2007.

2.   Claims 1-7, 10-20 and 22 have been amended.

3.   The object to Drawings and Specification is withdrawn in view of Applicant's filed amendment of specification and replacement drawing.

4.   The 35 U.S.C. 112 second paragraph rejection of claims 1-23 is withdrawn in view of the Applicant's amendment.

5.   The 35 U.S.C. 101 rejection of claims 16-17 is withdrawn in view of the Applicant's amendment.

6.   Claims 1-23 remain pending and have been examined.

Response to Arguments

7.   Applicant's arguments filed on 07/06/2007, in particular on pages 14-25, have been fully considered but they are not persuasive. For example:

- At page 20, third paragraph and page 22, third paragraph, the Applicant argues that the definition of the term "<u>program construct</u>" is a small piece of source code which makes up , for instance, a CASE statement or an IF statement or a DO loop in a program as disclosed in the specification. However the definition of the program construct has not been properly defined in the claims. Therefore, the Examiner can reasonably interpret the

program construct as a whole file of source code or a character of the source

code.

- At pages 19-20, the Applicant contends that Claims 1 and 10 are not being

  anticipated by Fogel because Fogel (CVS) compares copies of files to

  determine changes. By contrast, the claimed invention compares a changed

  construct in a file. However, as discussed above, because the limitation of

  program construct is not being defined in the claims, the Examiner interprets

  the size of construct is same as the file size. Therefore, as in previous office

  action, the prior art recited in page number 8-9 and 11-12, does disclose all

  the suggested the steps of amended claims 1 and 10.

- At page 22, the Applicant submits that Bloom does not teach the limitation of

  claim 11. As in Bloom, the first step in Fig.1, locating base module is not the

  equivalent of identifying a first construct. However, the Examiner respectfully

  disagrees. Because the same reason of definition about "construct" as

  addressed above, the base reference source programs which the Examiner

  cited, can be reasonably interpreted as the first construct.

- At page 22, the Applicant also argues that Bloom does not teach the

  amended step of (4) parsing M tokens of each one of the other constructs,

  where M=N; (5) comparing the M tokens of each one of the other constructs

  with the N tokens of the first construct. Because Bloom only discloses the

  comparison between a first version of a program and a second version of the

  program not between a first version of the program and a plurality of other

version of the program. However, as the applicant claimed in claim 11, page

7, "A method of determining if **two** or more constructs..."[emphasis added], in

the case of only has two constructs, the comparison has to perform between

these constructs. Therefore, Bloom does also teach the limitation of steps 4

and 5.

- At page 22. last paragraph, the Applicant further argues that nowhere in the

  disclosure of Bloom is there a reference to using weights in doing the

  comparison. However, Bloom discloses at Fig.1, steps , "test for next equal

  comparison of two consecutive lines of source codes" , step "Compare Result

  of Test" and step "Select Test producing Smallest area of change of change".

  These steps indicate that in order to select smallest area, the weights in doing

  the comparison has to be used. Therefore, the disclosure of Bloom still

  teaches the limitation of claim 11.

- At page 23, the Applicant argues that Bloom does not teach the all limitation

  of claim 20. However, as claim 20 is an article version of claimed method. For

  the same reason as discussed above. Bloom does disclose all the limitation

  of claim 20.

Claim Rejections - 35 USC § 112

8.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly
claiming the subject matter which the applicant regards as his invention.

9.　　Claim 10 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite

for failing to particularly point out and distinctly claim the subject matter which

applicant regards as the invention.

Claim 10:

Claim 10 recited the limitation "at least a minimal threshold size". However, it is

not clear what the "minimal threshold size" is. Because, the threshold is a specific

value not a range. It does not contain minimal/maximal value. For the purpose of

compact prosecution, the Examiner treats the "minimal threshold size" as –

threshold size--

Claim Rejections - 35 USC § 101

10.　　35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of
matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the
conditions and requirements of this title.

11.　　Claims 18 and 19 are rejected under 35 U.S.C. 101 because the claimed

invention is directed to non-statutory subject matter.

Claim 18:

Claims 18 claims an integrated development environment (IDE), which is a

software development tool, only consists of computer software program and can

be interpreted as computer program listings per se. Although, the Applicant

amends the claims to incorporate "when executed by a processor" for each of

elements, the integrated development environment is still considered as a

software program without being embedded in the computer readable storage

medium and executing by the processor. Such claimed computer programs do

not define any structural and functional interrelationships between the computer

program and other claimed elements of a computer, which permit the computer

program's functionality to be realized. Thus, they are not statutory.

See M.P.E.P. 2106.01 (I)

Claim 19:

Claim 19, depends from claim18, does not remedy the deficiencies as noted

above respectively, thus is also rejected under 35 U.S.C. 101 for the same

reasons.

Claim Rejections - 35 USC § 102

12.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(a) the invention was known or used by others in this country, or patented or described in a printed
publication in this or a foreign country, before the invention thereof by the applicant for a patent.

(b) the invention was patented or described in a printed publication in this or a foreign country or in public
use or on sale in this country, more than one year prior to the date of application for patent in the United
States.

(e) the invention was described in (1) an application for patent, published under section 122(b), by
another filed in the United States before the invention by the applicant for patent or (2) a patent
granted on an application for patent by another filed in the United States before the invention by the
applicant for patent, except that an international application filed under the treaty defined in section
351(a) shall have the effects for purposes of this subsection of an application filed in the United States

only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

12.   Claims 1-8 and 10 are rejected under 35 U.S.C. 102(b) as being anticipated by

<u>Fogel</u> (Fogel et al., Open Source Development with CVS, 3rd Edition).

Claim 1:

<u>Fogel</u> discloses an algorithm to improve efficiency of editing source code,

comprising

- recognizing that source code has been edited (see for example p.21, section

  "Finding Out What You (And Others) Did – update And diff", also see

  example, p.21, "cvs update" and related text);

- identifying a program construct having the edited source (see for example

  p.21, section "Finding Out What You (And Others) Did – update And diff", also

  see example, p.22-23, "cvs –Q diff –c" and related text);

- constructing a construct list of at least one other construct having derived

  and/or related code to the program construct (see for example, p.99, section

  "The checkoutlist File" and related description);

- comparing the similarity between the at least one other construct with the

  program construct having the edited source code (see for example p.21,

  section "Finding Out What You (And Others) Did – update And diff", also see

  example, p.22-23, "cvs –Q diff –c" and related text); and

- if, in response to comparing the at least one other construct with the program

  construct, a commonality between the at least one other construct and the

  program construct is found to be equal to or beyond a threshold of similarity,

then notifying a user responsible for the at least one other construct that the

source code of the program construct has been edited (see for example,

p.103, first paragraph, "By setting a watch on a file, a developer can have

CVS notify her if anyone else starts to work on that file. The notifications are

normally sent via email, although it is possible to setup other notification

methods").

Claim 2:

<u>Fogel</u> further discloses the efficiency algorithm of claim 1, wherein identifying the

program construct further comprises parsing tokens of the edited source code

(see for example p.21, section "Finding Out What You (And Others) Did – update

And diff", also see example, p.22-23, "cvs –Q diff –c" and related text).

Claim 3:

<u>Fogel</u> further discloses the efficiency algorithm of claim 1, wherein the step of

constructing a construct list further comprises determining that the at least one

other construct is of at least a threshold size for placement in the construct list

(see for example, p.18, section "Checking Out A Working Copy"; p.99, section

"The checkoutlist File" and related description).

Claim 4:

Fogel also discloses the efficiency algorithm of claim 3, further comprising parsing a sequence of tokens from each of a plurality of constructs of the at least threshold size (see for example p.21, section "Finding Out What You (And Others) Did – update And diff", also see example, p.22-23, "cvs –Q diff –c" and related text).

Claim 5:

Fogel further discloses the efficiency algorithm of claim 4, comparing the at least one other construct with the program construct further comprises comparing the parsed tokens of the edited source code with the parsed tokens of each of a plurality of constructs in the construct list (see for example p.21, section "Finding Out What You (And Others) Did – update And diff", also see example, p.22-23, "cvs –Q diff –c" and related text).

Claim 6:

Fogel further discloses the efficiency algorithm of claim 5, wherein the step of comparing the parsed tokens further comprises weighting the compared tokens to establish a degree of similarity (see for example, p.27, "see all the changes at once" and example of "floss$ cvs –Q diff –c" shows difference of all files in working directory comparing with files in repository).

Claim 7:

Fogel also discloses the efficiency algorithm of claim 6, further comprising summing the weights of the compared tokens to determine if the sum is equal to or beyond the threshold of similarity (see for example, p.27, "see all the changes at once" and example of "floss$ cvs –Q diff –c" shows difference of all files in working directory comparing with files in repository).

Claim 8:

Fogel further discloses the efficiency algorithm of claim 1, further comprising storing the construct list (see for example, p.99, section "The checkoutlist File": "If you look inside CVSROOT/, you'll see that working copies of the files exist side by side with their RCS revision files" and example).

Claim 10:

Fogel discloses an efficiency algorithm to improve efficiency of editing source code, comprising:

- recognizing that a source code has been edited (see for example p.21, section "Finding Out What You (And Others) Did – update And diff", also see example, p.21, "cvs update" and related text);

- identifying a program construct having the edited source code and parsing tokens of the edited source code (see for example p.21, section "Finding Out What You (And Others) Did – update And diff", also see example, p.22-23, "cvs –Q diff –c" and related text);

- constructing a construct list of at least one other construct of at least a minimal threshold size having related code by parsing a sequence of tokens from each of a plurality of constructs of the at least minimal threshold size (see for example, p.99, section "The checkoutlist File" and related description);

- comparing the parsed tokens of the edited source code with the parsed tokens of each of the plurality of constructs in the construct list, and weighting the compared tokens (see for example, p.27, "see all the changes at once" and example of "floss$ cvs –Q diff –c" shows difference of all files in working directory comparing with files in repository);

- summing the weights of the compared tokens to determine if the sum is equal to or beyond the threshold of similarity, and if so, then determining if a user responsible for the at least one other construct is to be notified (see for example, p.107, example of watch on cvs checkin process: "Triggered commit watch"); and

- storing the construct list (see for example, p.99, section "The checkoutlist File": "If you look inside CVSROOT/, you'll see that working copies of the files exist side by side with their RCS revision files" and example).

13.   Claims 11-15 and 20-23 are rejected under 35 U.S.C. 102(b) as being anticipated by Bloom (Delwin W. Bloom, US 3,711,863")

Claim 11:

Bloom discloses a method of determining if two or more constructs in a repository of source code in an integrated development environment are related and/or derived, said method comprising the steps of:

- identifying a first construct (see for example, Fig.1, step "Locate Base Module" and related text);

- parsing N tokens of the first construct, N being a positive integer (see for example, Fig.1, step "Load Source Code From Base Module Into First Working Buffer" and related text; also see Fig.2B step 38 "Compare seven words" and related text);

- identifying a plurality of other constructs in the repository (see for example, Fig.1, "Locate Module to be compared" and related text);

- parsing M tokens of each one of the other constructs, where M=N (see for example, Fig.1, step "Compare Source codes from Both Working Buffers Until Difference Between Codes is Found" and related text);

- comparing the M tokens of each one of the other constructs with the N tokens of the first construct (see for example, Fig.1, step "Test For Next Equal Comparison Of Two Consecutive Lines of Source Codes" and related text; also see Fig.2B step 38 "Compare seven words" and related text);

- determining a weight for each one of the N and M tokens based on name, type, and/or representation (see for example, Fig.1, step "Find Identical Symbolic Addresses" and related text);

- summing the weights of the N and M tokens (see for example, Fi.g1, step "Work Backwards From Identical Symbolic Addresses – Test For Non-Comparison" and related text);

- determining whether the sum of the weights of the M token meets or exceeds a threshold of similarity, the threshold of similarity being based on a percentage of the sum of the weights of the M tokens to the sum of the weights of the N tokens (see for example, Fig.1, step Compare Results of Tests" and related text; also see Fig.2c, step 62 :Calculate Total Lines of Change" and related text); and

- identifying each construct whose sum of the weights of the M tokens meets or exceeds the threshold of similarity as being related to the first construct (see for example, Fig.1, step "Select Test Producing Smallest Area of Change" and related text).

Claim 12:

Bloom further discloses the method of claim 11, wherein the step of identifying the first construct further comprises the step of identifying whether source code within which resides the first construct has been edited (see for example, Fig.1, step "Determine If Change is Deletion, Addition or Modification Line-By-Line" and related text).

Claim 13:

<u>Bloom</u> further discloses the method of claim 11, further comprising storing a pointer to each construct identified as being related to the first construct in a construct list of related construct (see for example, Fig.2C, step 66 "Set Pointer Based on Comparison Having Least Change", step "Set up Pointers" and related text).

Claim 14:

<u>Bloom</u> also discloses the method of claim 13, further comprising the step of identifying users responsible for each of the constructs in the construct list (see for example, Fig.2A step 12, "Read In Options Selected"; step 18 "setup Compare Parameters", step 20 "Initialize Search Flags" and related text).

Claim 15:

<u>Bloom</u> further discloses the method of claim 14, further comprising the step of offering notification to each user responsible for each one of the constructs in the construct list (see for example, Fig.1, steps "Determine If Change is Deletion, Addition or Modification Line-By-Line", "Print Change" and related text)

Claims 20-23:

Claims 20-23 are an article version to perform and realized the claimed methods as discussed in claims 11-15 above, wherein all claimed limitations have been

address and/or set forth in claims 11-15. Therefore, as the references teach all

the limitation of claims 11-15, they also teach he limitations of claims 20-23. Thus

they also would have been anticipated. (see for example, col.16, lines 11- col.20,

line 22)

Claim Rejections - 35 USC § 103

14.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set
forth in section 102 of this title, if the differences between the subject matter sought to be patented and
the prior art are such that the subject matter as a whole would have been obvious at the time the
invention was made to a person having ordinary skill in the art to which said subject matter pertains.
Patentability shall not be negatived by the manner in which the invention was made.

15.   Claims 9 and 16-19 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Fogel (Fogel et al., Open Source Development with CVS, 3rd Edition) in view

of CvsIn (CvsIn menu page)

Claim 9:

Fogel discloses the efficiency algorithm of claim 1, wherein the efficiency

algorithm is a machine-implemented process (see for example, p.69, section

"Getting And Installing CVS Under Windows" and related description; also see

p.70, last paragraph, "WinCvs"), but does not explicitly discloses the WinCvs in

an integrated development environment. However, CvsIn in the same analogous

art of source code version control disclose integration of CVS and IDE (see for

example, p.2, CvsIn- DevStudio Add-in for CVS and environment is Visual C++).

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to integrate CVS in the IDE by using CvsIn. One would have been motivated to do so to "have the working and operational tool Fast" as suggested by CvsIn (see for example, p.3, second paragraph)

Claims 16-17 and 18-19:

Claims 16-17 and 18-19 are two integrated development environment system versions of the claimed method/algorithm as discussed in claims 1-9 above, wherein all claimed limitations have been address and/or set forth in claims 1-9. Therefore, as the references teach all the limitation of claims 1-9, they also teach he limitations of claims 16-19. Thus they also would have been obvious by Fogel and CvsIn.

Conclusion

16.   The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

17.   Applicant's arguments with respect to claims rejection have been considered but are moot. Accordingly, the rejection of the claims over prior art in the previous office action is maintained in light of the necessitated additional clarification presented in this Office action and **THIS ACTION IS MADE FINAL**. See MPEP

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

18.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571) 270-1059 and Fax number is (571) 270-2059. The examiner can normally be reached on Monday-Thursday 8:00-15:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571- 272-1000.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

ZW

TUAN DAM
SUPERVISORY PATENT EXAMINER